Soccer Agents Using Inductive Learning with Hand-Coded Rules *

Hisayuki Sasaoka [1,2], Shunsuke Muraki [1], and Kenji Araki [3]

[1] Advanced Course of Production System Engineering,
Asahikawa National College of Technology,
Shunkodai 2-2, Asahikawa, 071-8142, JAPAN
[2] E-mail: sasa@asahikawa-nct.ac.jp
Tel. +81-166-55-8091

[3] Div. of Electronics and Information Engineering,
Graduate School of Engineering, Hokkaido University
Kita-13 Nishi-8, Kita-ku, Sapporo, 060-8628, JAPAN
E-mail: araki@media.eng.hokudai.ac.jp
Tel: +81-11-706-6534

Abstract – *This paper proposes a learning method for rules of soccer agents' actions in RoboCup Soccer. In general, soccer agents need to do high skilled actions in real time and they should have a mechanism to decide the actions. The soccer agents based on our method refer to some types of rules in order that they go into highly skilled actions. One type of them is the rules that are Hand-coded rules. Another type is the rules that are acquired using inductive learning and they are acquired from examples of actions, automatically. We describe the basic idea for them and indicate examples of rules.*

Keywords: multi agents, RoboCup Soccer, inductive learning, hand-coded rules, selecting actions.

1 Introduction

Soccer is one of the most popular sports in the world. Human soccer players need to make decisions for their actions and need to take some actions in real time on the games. For examples, dribble, passing, shoot, tackle and so on. Moreover, in order to get a shoot at the goal or to intercept adversary's passing, they should take highly skilled actions or actions beyond expectations. In order to take these excellent actions, we think that the team should develop strategies for the game and that the players of the team needs to have a lot of experience of playing soccer games and needs to have a good imagination for their actions.

This appearance is very similar to actions of soccer agents on simulation league in RoboCup soccer. They should make decisions for their actions on the simulation of soccer games. Recently actions of soccer agents have been improved from year to year [2]. On the games, soccer agents need to be a autonomous system. Then a lot of teams have reported the effectiveness of applying machine learning algorithm. Tsinghuaeolus [5] is the champion team of RoboCup 2001 and 2002. They have used reinforcement learning and their agents have evaluated and take adequate actions. The agents could show good performances in the competitions of simulation league in RoboCup Soccer. However, to improve their performances, engineers need to adjust some parameters for soccer agents and the operation requires a lot of knowledge about both soccer agents and RoboCup Soccer.

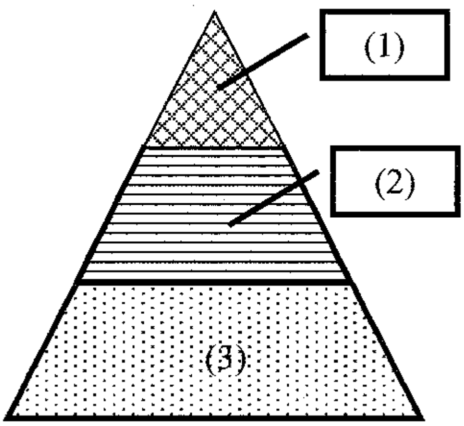
For the reason that soccer agents need to run autonomously, they require a lot of detailed rules for actions. A part of actions are prepared as basic commands for soccer games by the regulations of RoboCup. To hand-code all the detailed rules for these actions, it requires very huge costs. These hand-coded rules indicate the effectiveness for specific case. However the hand-coded rules have a low ability to adapt. Otherwise the system can acquire some rules using the method of machine learning. However it is very difficult for a system to acquire all the rules and to select the most adequate action by the only use of machine learning. The reason is that the method of machine learning needs a huge calculated amount for learning process. Moreover the acquiring rules could include some ineffective rules for soccer agents. The reason is that the system cannot know whether rules can work effectively or not until the rules can be applied. Therefore we propose a method for soccer agents using Inductive Learning with Hand-Coded Rules. In this method, we define two types of rules for actions of soccer agents. One of rules is called acquired rules. Another is called hand-coded rules. We explain the details of them in this paper.

We describe the basic idea of our method in section 2. Then the idea and example of hand-coded rules is indicated in section 3. Moreover we explain the process of acquiring rules from examples of action using inductive learning in section 4. In section, we explain the outline of process of selecting rules of actions. At last, in section 6, we show the conclusions of this paper.

2 Basic Idea

2.1 Rules for Actions of Agents

In this paper, we propose a method for acquiring and selecting rules for actions of soccer agents using Inductive Learning with Hand-Coded Rules. We call our method IL–HCR. We are interested in the interpretation of a person's capability for a knowledge acquisition and the application of it. In our previous researches, we have applied IL–HCR some applications of natural language processing [3], [4] and we have confirmed its effectiveness. In this research, this method applies to acquire and to select rules for actions of soccer agents.



(1) Gourp of rules obtained from reasoning
(2) Group of rules acquired using machine learning
(3) Group of rules given by other persons explicitly

Figure 1. Outline of Rules' Structure

Figure 1 illustrates our basic idea of a structure of rules for actions. We have considered that rules for actions can be classified into three groups. In this figure, hierarchy (1), (2), (3) means these three groups of rules and the structure of them.

For the human soccer, we consider that rules in Group (3) are comparable to instructions or strategies from coaches in their team. They are given by other persons explicitly. For example, in a lot of cases that the players should take defensive actions, the players followed the instructions from their coach and try to apply offside trap. In other case, a coach teaches some basic skills of soccer. For examples, the coach instructs players how to dribble or get a goal. We consider that engineers can give soccer agents a part of these rules explicitly. In this research, the rules in Group (3) are Hand-Coded Rules. We have developed the rules from books [1], [7]. The books present rules for actions of soccer players and strategies of games. Then the number of rules in Group (3) does not increase. The reason is that the entire rules are given at the beginning.

For the human soccer, we consider that the rules in Group (2) are comparable to empirical rules for actions of soccer players. Human soccer players that have a lot of experiences of soccer games can adapt to some situations in games and take adequate actions. In other cases, players can get improve their skill of soccer through trial and error during training. In our research, the rules in Group (2) are rules that are acquired from examples of actions using inductive learning. In this research, a part of the log-files of games are used as sources of extracting the rules. Then the number of rules increases every game. The increase of the number of rules can get the agents skillful. This is as well as human soccer player.

For the human soccer, we consider that the rules in Group (1) are comparable to rules that can generate creative actions. In rare cases, we can see that a player can take excellent actions that do not prepossess existing ideas and rules. The player might have a special concept or creativity for actions. The rules might be introduced in order to realize the special concept or creativity for actions. However we cannot implement them in this research. We are going to consider the details of this process and the rules and built into our agents in the near future. We think that agents can obtain these rules using reasoning. We also have a plan to consider the method to obtain the rules in Group (1) using reasoning.

2.2 Selecting Rules for Actions of Agents

In our method, when an agent takes actions, it refers to rules for actions in a dictionary of rules. The dictionary includes both Hand-Coded Rules and rules acquired from examples of actions using inductive learning. New acquired rules are added into the dictionary. Then the number of rules in the dictionary can increase. An agent needs to select an adequate rule from the dictionary.

When some rules of actions can be applied, agents need to select the most adequate rule referring to the values. The values mean the effectiveness of these rules. The values are a frequency of effective uses, a frequency of ineffective uses and a frequency of unknown uses. These three values are calculated after taking an action, automatically. This process is explained in section 5.

3 Hand-Coded Rules

This section presents the type of rules that are coded by hands. We have called the type of rules Hand-Coded Rules. In this paper, we have developed them from books [1], [7]. There are a lot of books for human soccer and if we consult them, it is very easy for us to code a lot of Hand-Coded Rules. One of reasons for introducing the Hand-Coded Rules is to resolve problems of shortage or data sparseness in source data of the learning process. For the reason of this, it is very difficult for us to prepare all examples of agents' actions. Engineers could refer to some books about human soccer and it is easy to code the rules of actions.

3.1 Hand-Coded Rules for Defense

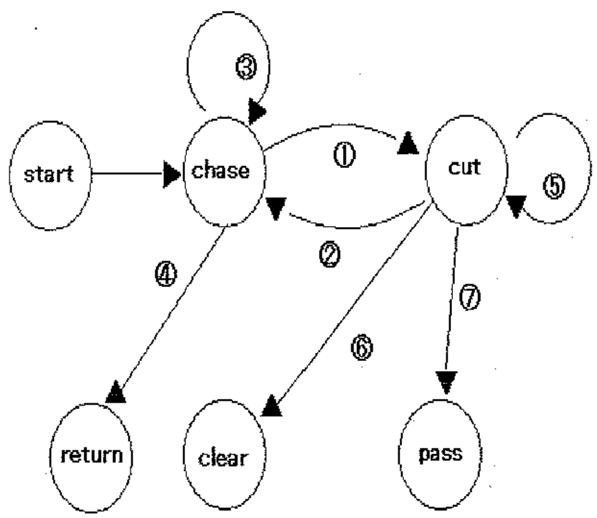


Figure 2. A State Transition Diagram of an Example of Hand-Coded Rules for Defense

Figure 2 shows a state transition diagram of an example of Hand-Coded Rules that the agents take actions for defense. Nodes in this figure indicate some actions of agents as the states. The numbers on the arcs indicate the condition for executing the actions. The action "chase" means that an agent gives chase to a ball. The action "cut" means that an agent cuts off a ball. The action "return" means that an agent goes back to the initial position of the agent. The action "clear" means that an agent gives a kick to a ball and tries to kick out from our own ground. The action "pass" means that an agent feeds a ball to other agent of our team. In short, an agent kicks a ball to an agent of their team.

In Figure 2, seven conditions and transitions are as follows:

1. the agent is going to try to cut the ball in the case that a distance between a ball and an agent of our team is within 1.

2. the agent is going to try to chase the ball in the case that a distance between a ball and an agent of our team is over 1.

3. the agent is going to try to cut the ball in the case that a distance between a ball and an agent of our team is within a range of defense.

4. the agent is going to try to return to their initial position in the case that a distance between a ball and an agent of our team is over 30.

5. the agent is going to try to cut the ball in the cae that a distance between a ball and an agent of our team is within 1.

6. the agent is going to try to clear the ball in the case that an agent of our team can see a ball and agents of our team and agents of adversary team.

7. the agent is going to try to do passing the ball to the agent of our team in the case that an agent of our team can see a ball and agents of our team and cannot see agents of adversary team.

When an agent takes an action "chase", in the case that the objects and the situation satisfy the requirement, that is, in the case that they satisfy the condition in state transition 1, the agent can take an action "cut". In the other case, the situation can satisfy the requirement, in short the condition in state transition 3, the agent can take an action "chase". Then the situation can satisfy the requirement, that is the condition in state transition 4, the agent can take an action "return". An agent can get the information of distance between the agent and objects from soccer server [6]. The objects include agents, ball, flag and so on. From the information, the system can judge whether the situation can satisfy the requirements. In addition to this action, in the action "cut", when objects and its situation satisfy the condition in state transition 7, it can take the actions "pass". Moreover, when objects and its situation satisfy the condition in state transition 6, it can take the actions "clear". Then when objects and its situation satisfy the condition in state transition 2, it can take the actions "chase".

In this example of rules, the action can take the only basic actions. Then we should consider the advanced rules to take more highly skilled actions. For examples, agents apply an offside trap or coordination actions for defense.

3.2 Hand-Coded Rules for Offense

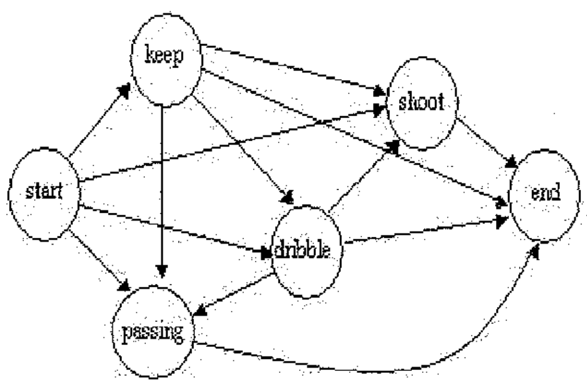


Figure 3. A State Transition Diagram of an Example of Hand-Coded Rules for Defense

Figure 3 illustrates a state transition diagram of an example of Hand-Coded Rules that the agents take actions for offense. Nodes in this figure also indicate some actions of agents. The action "passing" means that an agent feeds a ball to other agent of our team. In short, an agent kicks a ball to an agent of their team. The action "keep" means that an agent puts a ball within a range of its control and holds the ball. The action "dribble" means that an agent dribbles with a ball. In short, an agent kicks a ball with a little power. The action "shoot" is means an agent shoots. In short, an agent kicks a ball to the goal.

An agent can take from one action to another action as well as the state transition diagram of rules for defense. When an agent takes an action "keep", in the case that the situation can satisfy the requirement, the agent can take an action "passing". In the other case, the situation can satisfy the requirement that is different from the requirement for the action "passing", the agent can take an action "dribble". Moreover the situation can satisfy the requirement that is different from the requirement for the action "passing" and the action "dribble", the agent can take an action "shoot". When an agent and its situation satisfy these requirements, it can take the action as well as the action "keep". In this figure, some conditions for actions are overleaped.

Moreover we need to consider how an agent that is not near the ball should take actions. For human soccer, when a player does not keep or kick a ball, it is very important to take an adequate action and to move to a good position. Then we have a plan to code rules for the situation in the near future.

4 Rules Acquired by Inductive Learning

4.1 Outline

A process based on inductive learning composes mainly of two processes. One of components is an acquiring process that extracts rules from examples of actions. The soccer server program records log-files of all agents' actions [6]. In this process, the agents use a part of the log-files as sources of acquired rules. In this process, they refer to the information of character strings in log-files. We defined a source pair of rules as a pair of an action of adversary team's agents and an action of our team's agents. The system extracts the acquired rules from two source pairs of rules, automatically. The extracted rules include a pair of an action of adversary agents and an action of our teams.

On the other hand, another component is a feedback process. In a feedback process, a system learns confidences for the rules from results of using the rules. Both acquired rules and hand-coded rules are used by mixture. However these rules include differences in terms of confidences derived from their history of using. In a case that a rule would be used in a correct or an effective action for their team, the rule could be considered as one of effective rules. In an opposite case, the rules could be considered as one of ineffective rules. The system could calculate the number of confidence from the history of using rules.

4.2 Acquiring Process

The agents can use log-files of games as sources of extracting rules. Soccer Server records all commands of actions of agents in one log-file. The log-files include a lot of unnecessary information. The reason of this is that we need to execute some pre-processes. One of them is to remove some line of command of actions before the beginning of games. Another process is to arrange actions of agents according to the identical number of agents. Then the agents can extract some rules from the sources with the information of character strings. It extracts common and different parts of character strings. Each part can be new rule and add them into the dictionary of rules.

4.3 Feedback Process

In order that agents can evaluate rules, when it refers to one rule, it calculates some values that are attached to the rules. The values are calculated after taking an action. The values are as follows:

1. a frequency of effective uses
2. a frequency of ineffective uses
3. a frequency of unknown uses

A value "a frequency of effective uses" is the number of times in the case of taking the effective action by selecting the rule. A value "a frequency of ineffective uses" is the number of times in the case of taking the ineffective action by selecting the rule. A value "a frequency of unknown uses" is the number of times in the case that selecting rules lead to take the action that is not judged neither an effective nor an ineffective action.

In order to judge whether an action is effective or not exactly, an engineer should judge the action in every step of game. However it costs very huge. Then we define some standard of the judgments and an agent can judge the action autonomously. Examples of standards of an effective action "kick" are as follows:

1. a ball gets into the goal of adversary team
2. a ball keep within the rage of its control
3. a ball goes out the its side ground

In the case that some objects and the situation after taking an action satisfy the standards of them, the system adds 1 into its "a frequency of effective uses".

On the other hand, examples of an effective action "kick" are as follows:

1. a playmode of soccer game is changed
2. a ball moves to the rage that adversary agent can control

In the case that some objects and the situation after taking an action satisfy the standards of them, the system adds 1 into its "a frequency of ineffective uses".

Moreover, in taking one action, in the case that the objects and the situation can satisfy neither standards of an effective action nor standards of an effective action, the system adds 1 into its "a frequency of unknown uses".

5 Process of Selecting Rules

When an agent needs to select the most adequate rule from other rules, it refers to some values that are attached to the rules of actions. The values are "a frequency of effective uses", "a frequency of ineffective uses" and "a frequency of unknown uses". The sequence of referring the values is as follows:

1. a frequency of effective uses
2. a frequency of ineffective uses
3. a frequency of unknown uses

In the case that one frequency of effective uses is larger than other frequencies of effective uses, the agent can consider as more effective rules than other rules. Conversely, in the case that one frequency of ineffective uses is smaller than other frequencies of ineffective uses, the agent can consider as less ineffective rules than other rules. Then our system selects rule that has a larger frequency of unknown uses. On the other hand, when all the values are equal, the agents select the rule that is newer than other rules. The reason of this selection is the heuristics.

6 Conclusions

This paper describes the basic idea of our method for acquiring and selecting rules of action soccer agents using Inductive Learning with Hand-coded Rules. We have called our method IL–HCR. Then we indicate some examples of Hand-Coded Rules for actions of soccer agents. Then it describe outline of the process of acquiring rules.

We have a plan to do evaluation experiments and to confirm the effectiveness of this method. Moreover we have to consider about the size and the number of action rules and discuss about a process in a case that our system do not have any adequate rules. In addition of this, we should add some Hand-Coded Rules into the dictionary of rules.

References

[1] FC Tokyo, "*Kohgeki* Soccer – *Goal wo Mezase !*", *Ikeda Shoten*, Tokyo, 2001.

[2] Hidehisa Akiyama, "The Tendency of RoboCup Soccer Simulation League", *Journal of the Japanese Society for Artificial Intelligence*, vol.17, no.6 pp. 708 – 713, 2002.

[3] Hisayuki Sasoka, Kenji Araki, Yoshio Momouchi, and Koji Tochinai, "Evaluation of Spoken Language Translation Mehod Using Machine-acquiring Rules and Hand-made Rules", *Information Processing Society of Japan , SIG Notes* NL 145-2, pp. 9 – 14, 2001.

[4] Hisayuki Sasoka, Kenji Araki, Yoshio Momouchi, and Koji Tochinai, "Evaluation of Prediction Method of Target Words Using Inductive Learning", *Proceedings of Artificial Intelligence and Soft Computing 1998, IASTED,* pp. 183 – 186, 1998.

[5] Jinyi Yao, Jiang Chen, Yunpeng, and Shi Li, "Architecture of TsinghuAeolus", RoboCup 2001: *Robot Soccer World Cup V*, Springer Verlag, pp. 491 – 494, 2002.

[6] Mao Chen, Klaus Dorer, Ehsan Foroughi, Fredrik Heintz, Zhan Xiang Huang, Spiros Kapetanakis, Kostas Kostiadis, Johan Kummeneje, Jan Murray, Itsuki Noda, Oliver Obst, Pat Riley, Timo Steffens, Yi Wang and Xiang Yin, "User Manual RoboCup Soccer Server for Soccer Server Version 7.07 and later", http://sserver.sourceforge.net/

[7] *NISSAN* F.C., "Soccer: *Senjyutsu Hen*", *Ikeda Shoten*, Tokyo, 1994.